

Tầng liên kết

- 5.1 Giới thiệu và dịch vụ
- 5.2 Sự phát hiện và Sửa lỗi
- 5.3 Các giao thức đa truy cập
- 5.4 Đánh địa chỉ tầng-Liên kết
- 5.5 Ethernet
- 5.6 Các bộ chuyển mạch tầng-liên kết
- 5.7 PPP

Địa chỉ MAC và ARP

□ địa chỉ IP 32-bit:

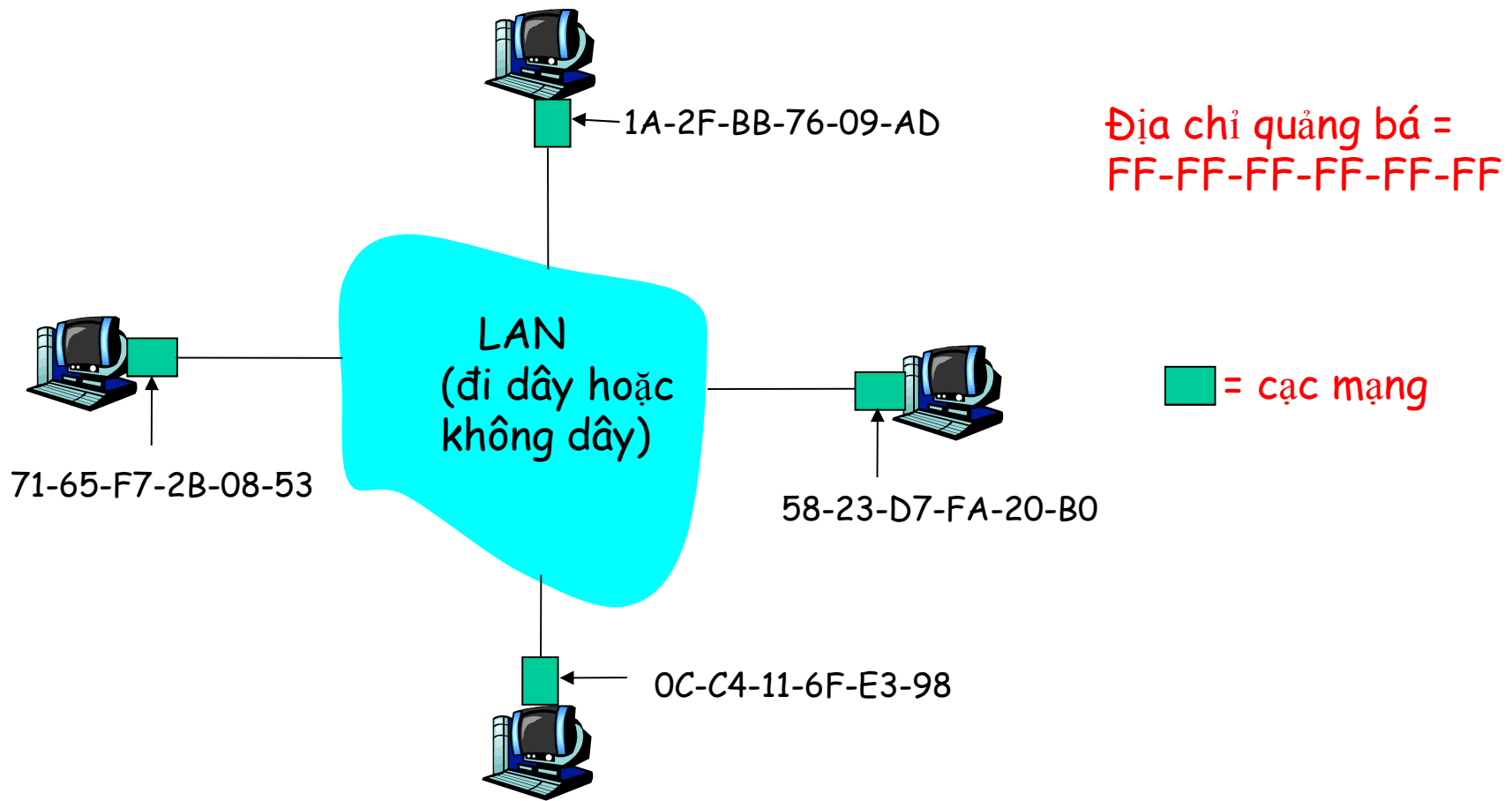
- địa chỉ *tầng-mạng*
- dùng để gửi gói tin tới mạng IP đích

□ Địa chỉ MAC (hay LAN hay vật lý hay Ethernet):

- vai trò: *chuyển khung từ giao diện này tới giao diện kết nối-vật lý khác (trong cùng mạng)*
- địa chỉ MAC 48 bit (cho hầu hết LANs)
 - được gán cứng vào NIC ROM, đôi khi có thể thay đổi được bằng phần mềm

Địa chỉ LAN và ARP

Mỗi cạc mạng LAN có một địa chỉ LAN độc nhất (ko trùng nhau)



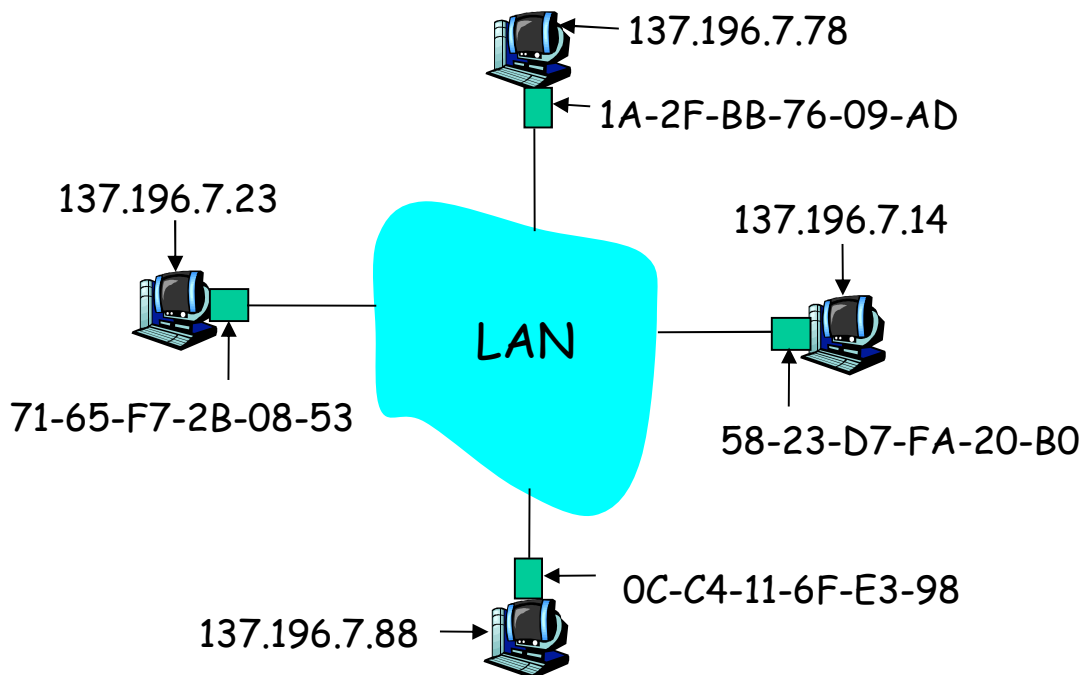
Địa chỉ LAN (tt)

- ❑ sự phân phối địa chỉ MAC được quản lí bởi IEEE
- ❑ các nhà sản xuất mua một phần của không gian địa chỉ MAC (để đảm bảo tính độc nhất)
- ❑ địa chỉ MAC phẳng (bình đẳng, đồng dạng) → tính di động
 - có thể di chuyển các mạng từ một LAN sang LAN khác
- ❑ địa chỉ phân lớp IP **KHÔNG di động**
 - địa chỉ IP phụ thuộc vào mạng con mà nó gắn vào

ARP: Giao thức phân giải địa chỉ

(Address Resolution Protocol)

Câu hỏi: làm sao để xác định địa chỉ **MAC** của B nếu biết địa chỉ **IP** của B?



- Mỗi nút (máy, bđt) trên LAN có 1 bảng **ARP**
- bảng ARP: các ánh xạ địa chỉ IP/MAC của một vài nút trong LAN

< địa chỉ IP; địa chỉ **MAC**; TTL >

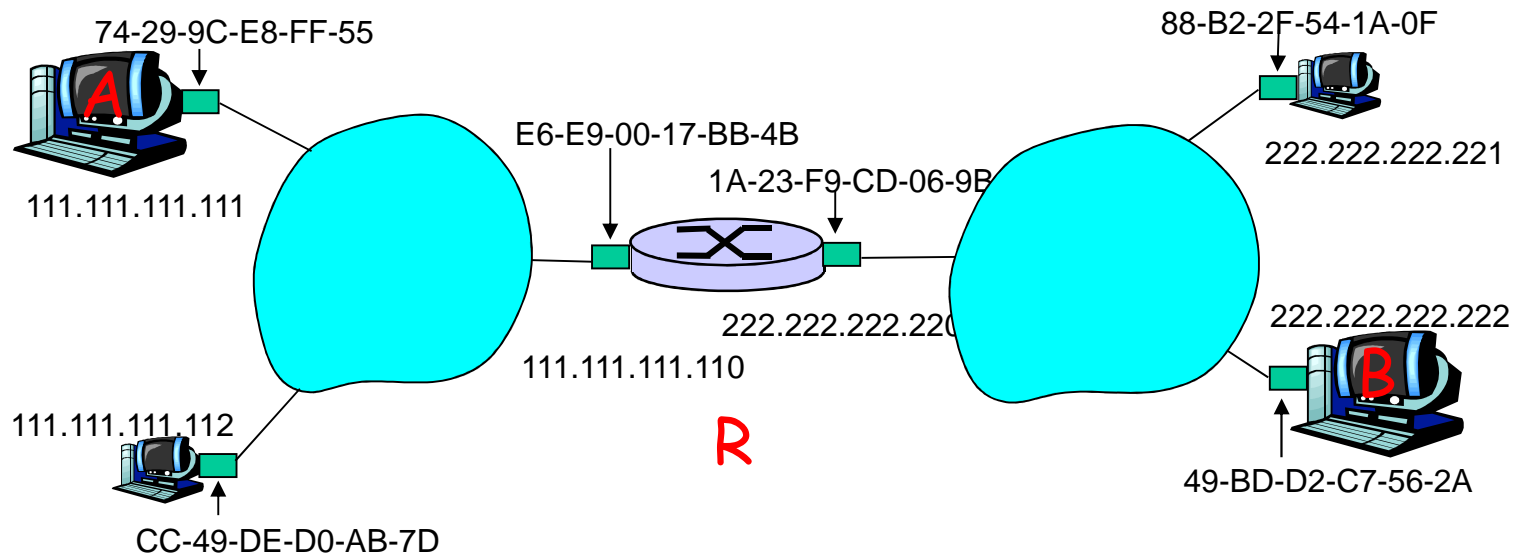
- thời gian sống TTL (Time To Live): thời gian tồn tại của một ánh xạ trong bảng ARP, sau t/g này ánh xạ sẽ bị xóa đi (thường là 20 phút)

Giao thức ARP: cùng LAN (mạng)

- ❑ A muốn gửi gói tin cho B, và địa chỉ MAC của B không nằm trong bảng ARP của A.
- ❑ A **quảng bá** gói truy vấn ARP, chứa địa chỉ IP của B
 - địa chỉ MAC đích = FF-FF-FF-FF-FF-FF
 - tất cả các máy trên LAN đều nhận truy vấn ARP
- ❑ B nhận được gói truy vấn ARP, phản hồi cho A với địa chỉ MAC của nó (B)
 - khung được gửi tới địa chỉ MAC của A (**gửi-1-đích**)
- ❑ Một bản lưu cặp địa chỉ IP-MAC được giữ trong bảng ARP của A cho đến khi hết thời gian sống của nó.
 - trạng thái mềm: t/tin sẽ bị xóa khỏi bảng ARP nếu không được làm mới
- ❑ ARP "cắm-và-chạy":
 - các nút tạo ra bảng ARP của chúng mà *không có sự can thiệp từ phía quản trị viên của mạng*

Đánh địa chỉ: định tuyến tới LAN khác

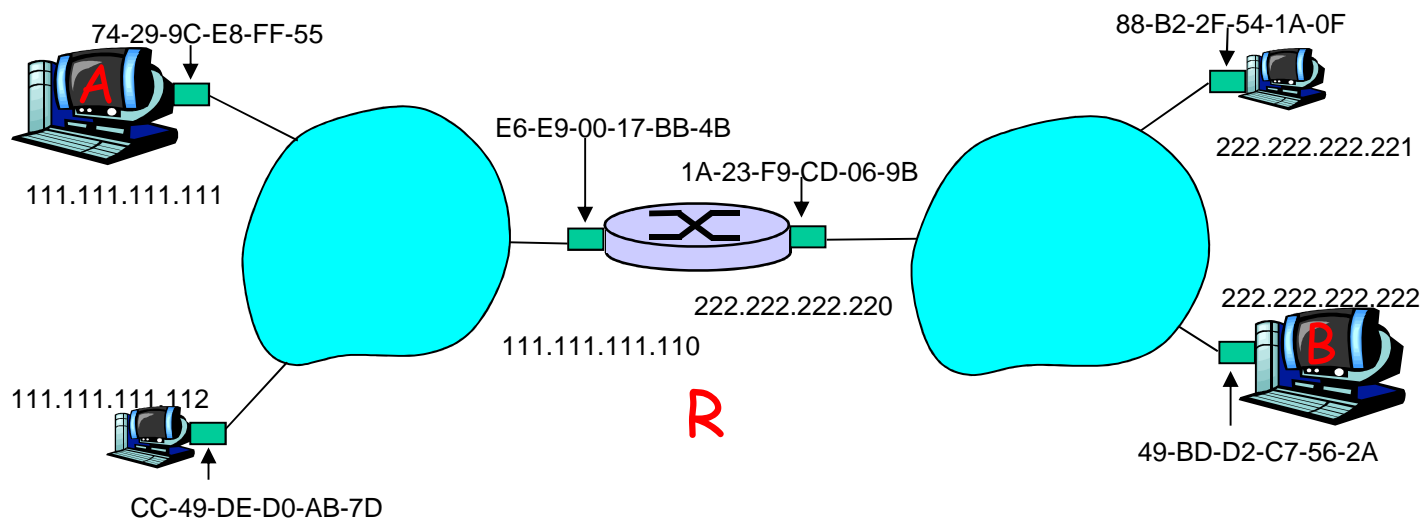
các bước: **gửi gói tin từ A sang B thông qua R**
giả sử A biết IP của B



- hai bảng ARP trong bộ R, một cho mỗi mạng IP (LAN)

- ❑ A tạo ra gói IP với nguồn A, đích B
- ❑ A sử dụng ARP để lấy địa chỉ MAC của R với IP là 111.111.111.110
- ❑ A tạo khung tầng-liên kết với địa chỉ đích là đ/c MAC của R, khung chứa gói tin IP A-tới-B
- ❑ NIC A gửi khung
- ❑ NIC R nhận khung
- ❑ R gỡ bỏ gói IP từ khung Ethernet, thấy nó gửi cho B
- ❑ R sử dụng ARP để lấy địa chỉ MAC của B
- ❑ R tạo ra khung chứa gói tin IP A-tới-B ,gửi cho B

Đây **thực sự** là một ví dụ quan trọng - hãy chắc là bạn nắm rõ!



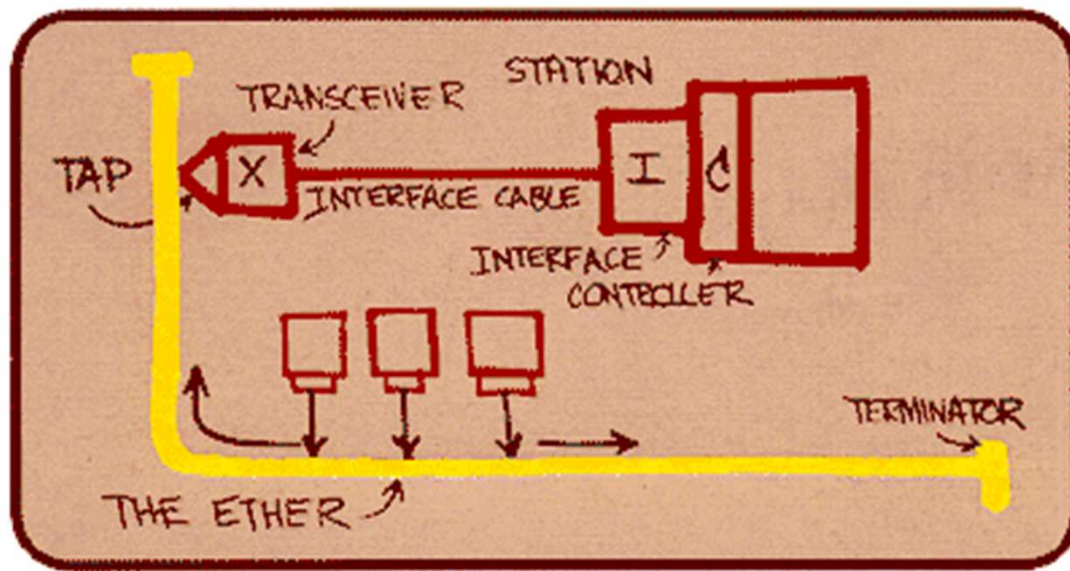
Tầng liên kết

- 5.1 Giới thiệu và dịch vụ
- 5.2 Sự phát hiện và Sửa lỗi
- 5.3 Các giao thức đa truy cập
- 5.4 Đánh địa chỉ tầng-Liên kết
- 5.5 Ethernet
- 5.6 Các bộ chuyển mạch tầng-liên kết
- 5.7 PPP

Ethernet

công nghệ "thống trị" của LAN đi dây:

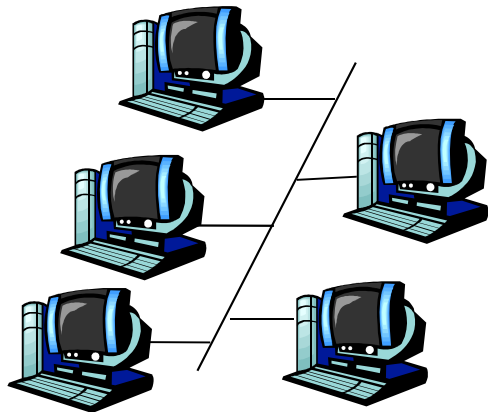
- ❑ rẻ, \$20 cho mỗi NIC
- ❑ công nghệ LAN đầu tiên được dùng rộng rãi
- ❑ đơn giản hơn, rẻ hơn LAN dùng thẻ và ATM
- ❑ theo kịp nhịp tăng tốc: 10 Mbps - 10 Gbps



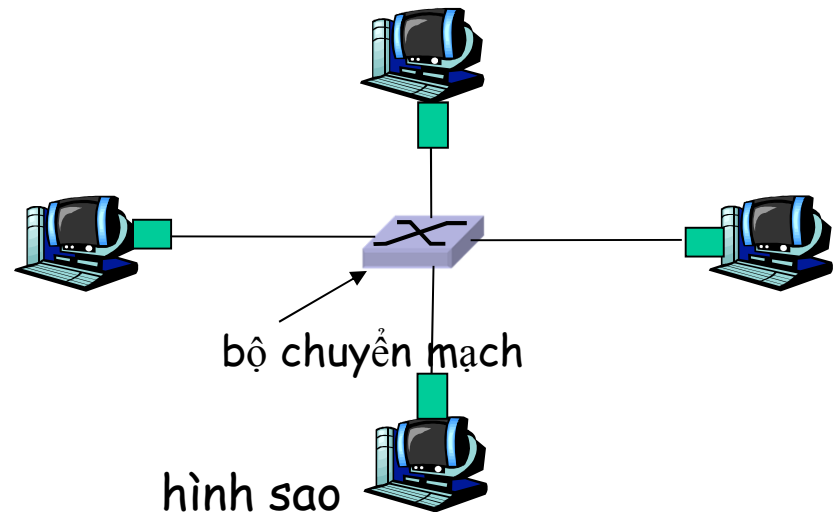
bản phác thảo
Ethernet của Metcalfe

Sơ đồ hình Sao

- ❑ sơ đồ buýt phổ biến suốt những năm 90
 - tất cả nút trong cùng một miền đưng độ (có thể đưng độ với với nhau)
- ❑ ngày nay: sơ đồ Sao chiếm ưu thế
 - *bộ chuyển mạch* hoạt động tại trung tâm
 - mỗi "nhánh" chạy một giao thức Ethernet riêng lẻ (nút không va chạm với nhau)



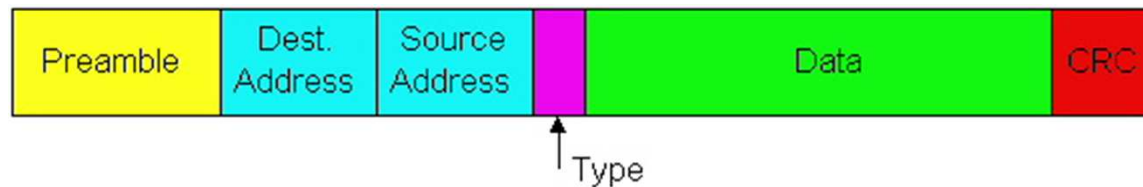
bút: cáp đồng trục



hình sao

Cấu trúc khung Ethernet

Nic gửi đóng gói gói IP (hoặc là gói tin của giao thức tầng khác) vào **Khung ethernet**



Phần khởi đầu:

- ❑ 7 byte với mẫu 10101010 theo sau bởi 1 byte với mẫu 10101011
- ❑ sử dụng để đồng bộ hóa tốc độ đồng hồ của người gửi với người nhận.

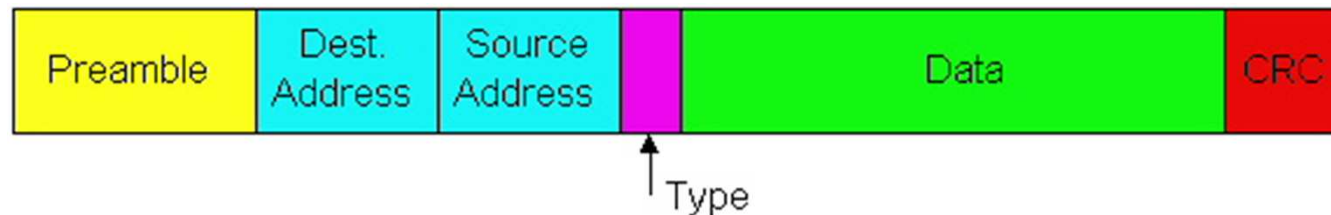
Cấu trúc khung Ethernet (tt)

❑ Địa chỉ: 6 bytes

- nếu NIC nhận được khung với đúng địa chỉ MAC của nó hoặc là địa chỉ phát tán rộng (vd gói tin ARP), nó sẽ đẩy dữ liệu trong khung lên giao thức tầng mạng
- ngoài ra, NIC loại bỏ khung

❑ Loại: xác định giao thức tầng cao hơn (hầu hết là IP nhưng thỉnh thoảng có những g/t khác, vd, Novell IPX, AppleTalk)

❑ CRC: kiểm tra tại người nhận, nếu có lỗi, khung sẽ bị bỏ



Ethernet: không tin cậy, không kết nối

- ❑ **không kết nối**: không có bắt tay giữa các NIC gửi và nhận
- ❑ **không tin cậy**: NIC nhận không gửi ACK hoặc là NACK cho NIC gửi
 - luồng gói tin truyền tới tầng mạng có thể có chỗ gián đoạn (các gói tin bị mất)
 - các chỗ gián đoạn có thể được lấp đầy nếu sử dụng TCP
 - ngoài ra, sử dụng sẽ thấy các chỗ gián đoạn này
- ❑ Giao thức MAC của Ethernet: **CSMA/CD** không chia-ô

Giải thuật CSMA/CD Ethernet

1. NIC nhận được gói tin từ tầng mạng, tạo ra khung
2. Nếu NIC thấy kênh truyền rỗi, bắt đầu truyền khung. Nếu NIC thấy kênh bận, đợi đến khi kênh rỗi, sau đó truyền
3. Nếu NIC gửi toàn bộ khung đi mà không phát hiện ra sự truyền tải nào khác, NIC hoàn thành việc gửi khung!
4. Nếu NIC phát hiện sự truyền tải khác trong khi đang truyền: hủy bỏ và gửi tín hiệu nghẽn (jam signal)
5. Sau khi hủy bỏ việc gửi, NIC bước vào **thoái lui hàm mũ - exponential backoff**: sau lần đụng độ thứ m , NIC chọn K ngẫu nhiên từ $\{0, 1, 2, \dots, 2^m - 1\}$. NIC chờ $K \cdot 512$ t/gian بیت, quay lại bước 2

CSMA/CD Ethernet (tt)

Tín hiệu tắc nghẽn: đảm bảo rằng tất cả các người gửi khác biết về sự ùng độ; 48 bits

T/g bit: .1 microsec cho mạng Ethernet 10 Mbps; với $K=1023$, thời gian chờ vào khoảng 50 msec

Xem/tương tác với vi mã Java trên Web AWL: rất khuyến khích !

thoái lui hàm mũ:

- **Mục tiêu:** thay đổi thời gian chờ truyền lại cho phù hợp với tải hiện tại
 - tải nặng: thời gian chờ ngẫu nhiên sẽ dài hơn
- ùng độ đầu tiên: chọn K từ $\{0,1\}$; độ trễ là $K \cdot 512$ t/g bit
- ùng độ lần 2: chọn K từ $\{0,1,2,3\}$...
- sau va chạm lần 10, chọn K từ $\{0,1,2,3,4,...,1023\}$

Hiệu suất CSMA/CD

- $T_{\text{lan truyền}}$ = độ trễ lan truyền tối đa giữa 2 nút LAN
- $t_{\text{truyền tải}}$ = thời gian để truyền tải khung lớn nhất

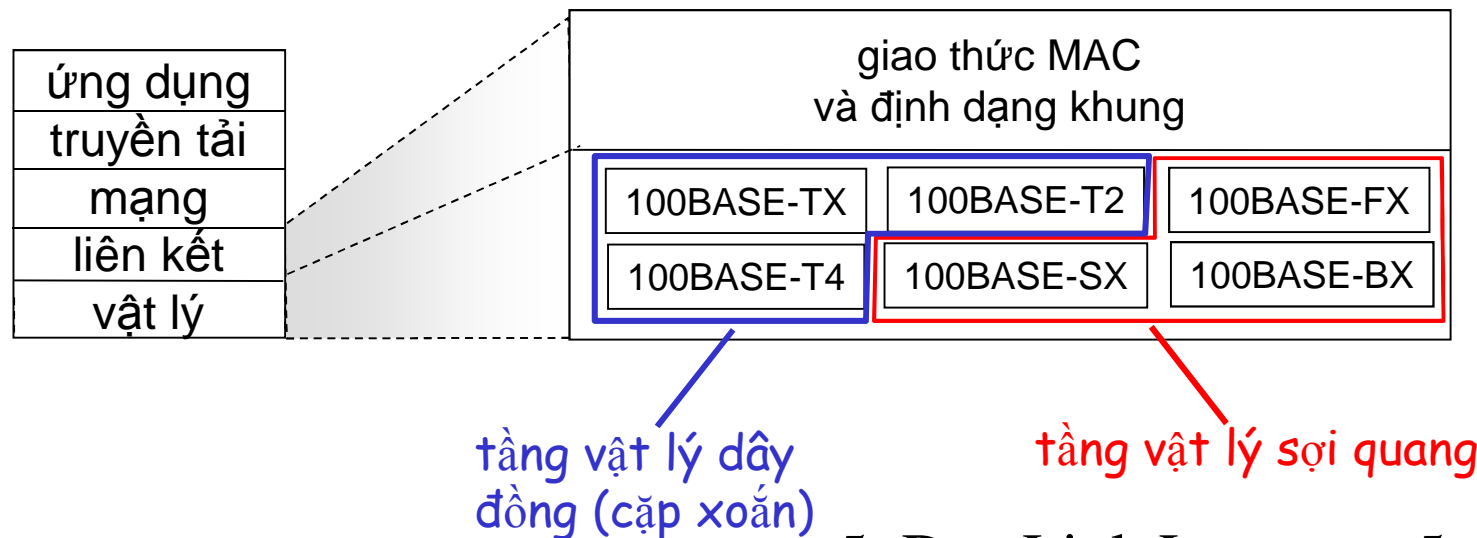
$$Hieusuat = \frac{1}{1 + 5t_{\text{lantruyen}} / t_{\text{truyentai}}}$$

- Hiệu suất tiến tới 1 khi
 - $t_{\text{lan truyền}}$ tiến tới 0
 - $t_{\text{truyền tải}}$ tiến tới vô cùng
- Hiệu suất tốt hơn ALOHA: và đơn giản, rẻ, không tập trung!

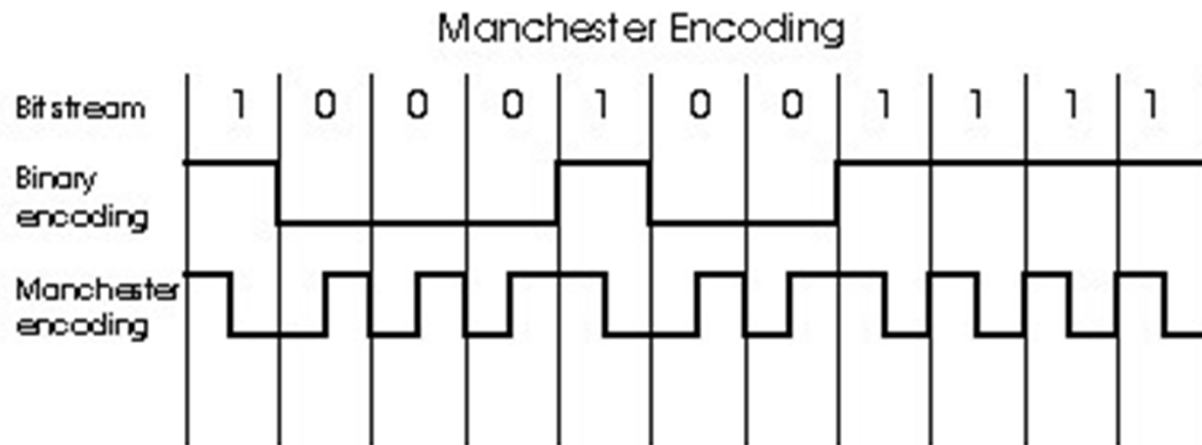
Chuẩn Ethernet 802.3: Tầng Liên Kết và Vật Lý

□ *nhiều* chuẩn Ethernet khác nhau

- giao thức MAC và định dạng khung phổ biến
- vận tốc khác nhau: 2 Mbps, 10 Mbps, 100 Mbps, 1Gbps, 10G bps
- môi trường vật lý khác nhau: cáp quang, cáp TH



Chuyển mã Manchester



- ❑ sử dụng trong 10BaseT
- ❑ mỗi bit có một sự chuyển đổi
- ❑ cho phép các đồng hồ ở phía nhận và gửi đồng bộ hóa với nhau
 - không cần đồng hồ tập trung, tổng quát cho các nút!
- ❑ Nhưng, đây là vấn đề của tầng-vật lý!

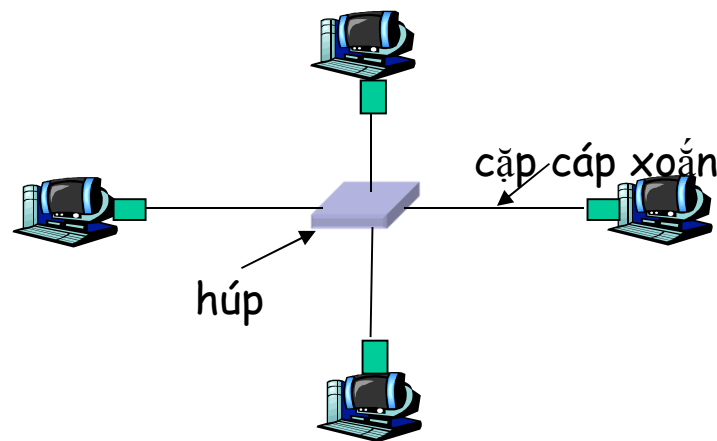
Tầng liên kết

- 5.1 Giới thiệu và dịch vụ
- 5.2 Sự phát hiện và Sửa lỗi
- 5.3 Các giao thức đa truy cập
- 5.4 Đánh địa chỉ tầng-Liên kết
- 5.5 Ethernet
- 5.6 Các bộ chuyển mạch tầng-liên kết
- 5.7 PPP

Hubs

... những bộ lặp tầng vật lý ("cân"):

- các bit vào trên 1 cổng và đi ra **tất cả** các cổng khác với cùng tốc độ
- tất cả các nút kết nối tới 1 húp có khả năng đụng độ với nhau
- không nhớ tạm khung
- không có CSMA/CD tại húp: NIC của máy phát hiện ra đụng độ

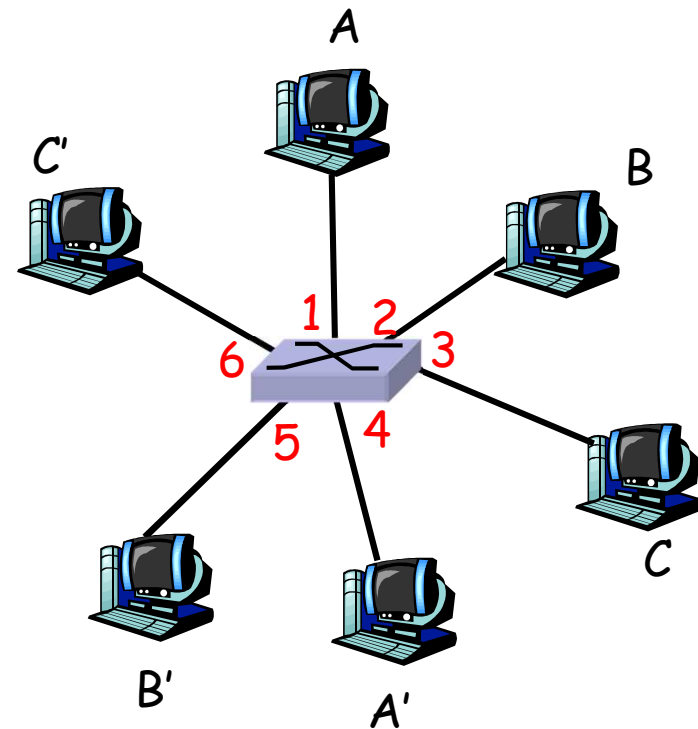


Bộ chuyển mạch (Switch)

- ❑ thiết bị tầng-liên kết: thông minh hơn húp, thực hiện vai trò *chủ động*
 - lưu, chuyển tiếp các khung Ethernet
 - kiểm tra địa chỉ MAC của các khung tới, chuyển tiếp khung có *lựa chọn* tới một hoặc nhiều đường ra, sử dụng CSMA/CD để truy cập đoạn mạng
- ❑ *trong suốt*
 - các máy sẽ không biết có sự tồn tại của các BCM
- ❑ *cắm-và-chạy, tự-học*
 - BCM không cần phải được cấu hình

BCM: cho phép *nhiều* sự truyền tải đồng thời

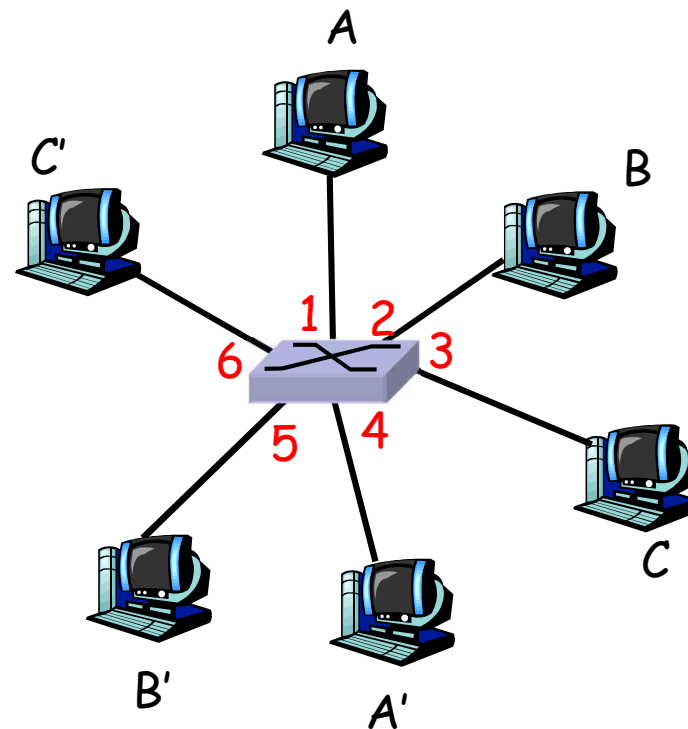
- ❑ máy có liên kết chuyên biệt, trực tiếp tới bcm
- ❑ bcm nhớ tạm các gói tin
- ❑ g/thức Ethernet được dùng trên *mỗi* liên kết vào, nhưng không có độn độ; song công
 - mỗi liên kết nằm trong miền độn độ riêng của mình nó
- ❑ *chuyển mạch*: A-tới-A' và B-tới-B' đồng thời, không có độn độ
 - ko thể xảy ra với các húp "câm"



BCM với 6 cổng
(1,2,3,4,5,6)

Bảng chuyển mạch

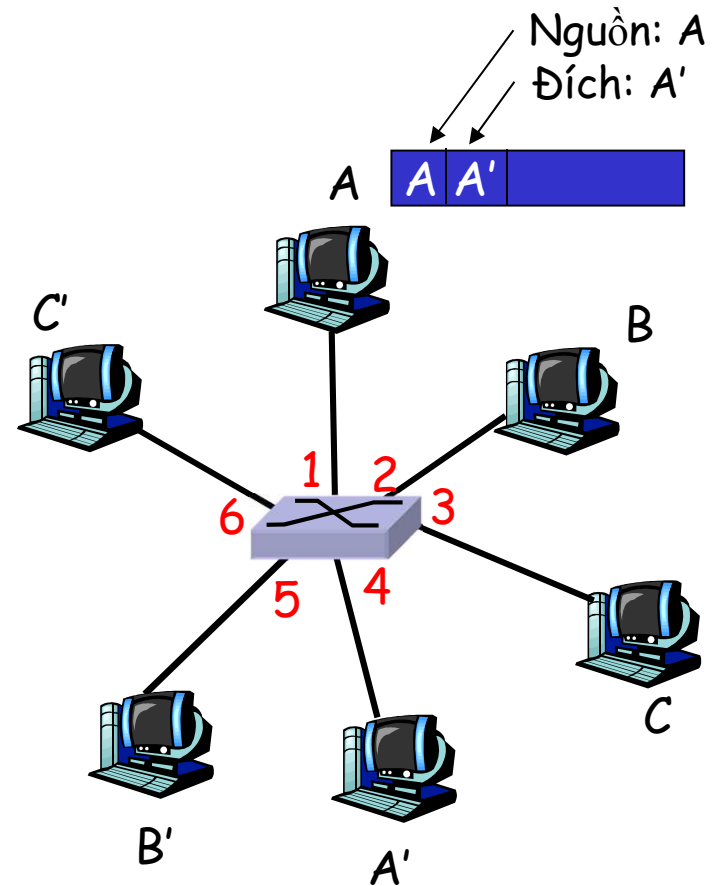
- ❑ H: làm sao BCM biết được là A' có thể tới được qua cổng 4, B' có thể tới được qua cổng 5?
- ❑ Đ: mỗi bcm có một **bảng chuyển mạch**, mỗi mục là:
 - (đ/c MAC của máy, cổng để tới máy đó, tem t/gian)
- ❑ giống như bảng định tuyến?
- ❑ H: làm sao các mục được tạo ra, duy trì trong bảng c/mạch?
 - giống giao thức định tuyến?



BCM với 6 cổng
(1,2,3,4,5,6)

BCM: tự-học

- bcm *học được* máy nào có thể tới được qua cổng nào
 - khi nhận được khung, bcm "học" vị trí của n/gửi: đoạn mạng LAN từ đó khung đi vào
 - lưu lại cặp ng/gửi, vị trí trong bảng chuyển mạch



đ/c MAC	cổng	TTL
A	1	60

*Bảng chuyển mạch
(ban đầu trống)*

BCM: lọc/chuyển tiếp khung

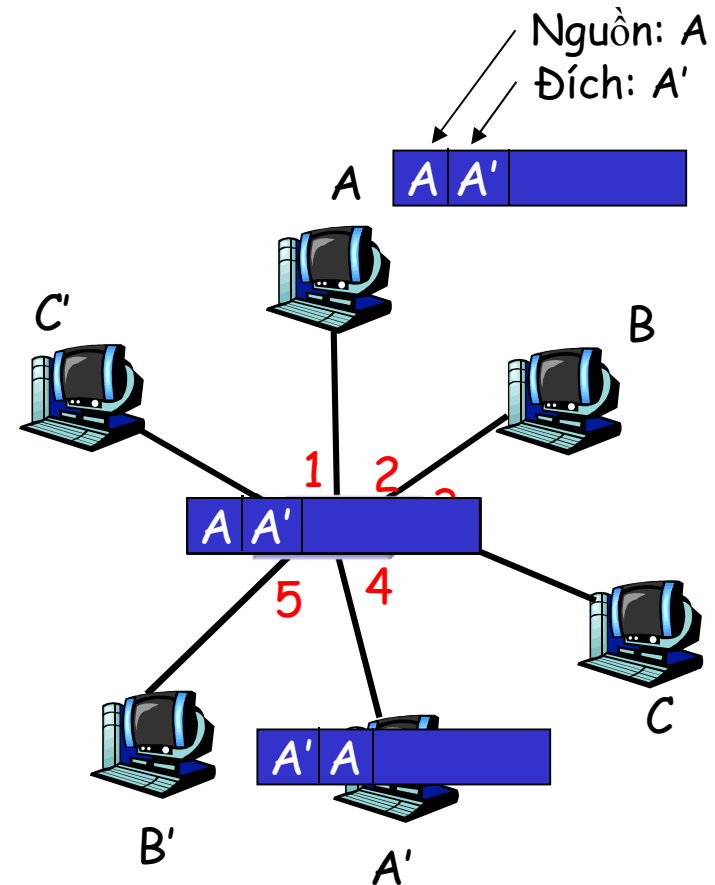
Khi một khung tới bcm:

1. lưu lại cổng liên kết với máy gửi
2. tìm chỉ mục trong bảng c/m sử dụng địa chỉ MAC đích
3. **if** tìm thấy mục của đích
 then {
 if nếu đích nằm trên cùng đoạn mạng mà từ đó khung tới
 then bỏ khung
 else chuyển tiếp khung tới cổng tương ứng trong bảng c/m
 }
 else gửi tràn

chuyển tiếp tới tất cả các cổng, ngoài trừ cổng mà từ đó khung tới

Tự-học, chuyên tiếp: Ví dụ

- không biết đích:
gửi tràn
- vị trí đích A đã biết:
gửi có lựa chọn

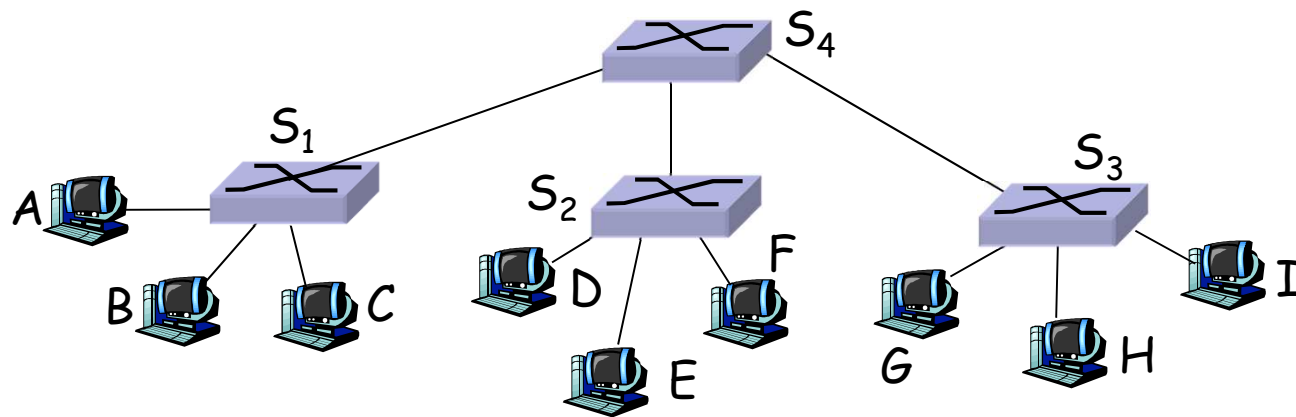


đ/c MAC	cổng	TTL
A	1	60
A'	4	60

*bảng c/m
(ban đầu trống)*

Các BCM kết nối với nhau

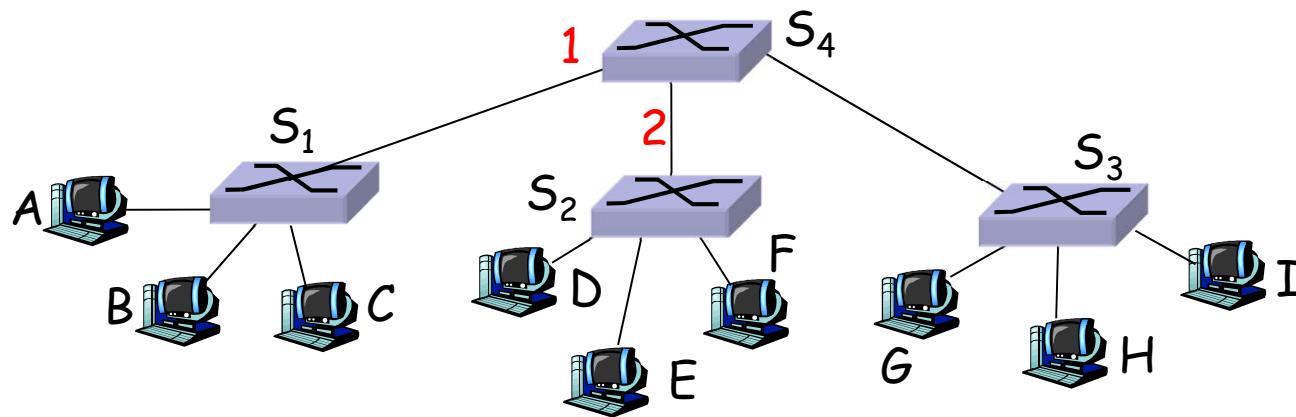
- các bcm có thể kết nối với nhau



- H: gửi từ A tới G - làm sao S₁ biết để chuyển khung tới G thông qua S₄ và S₃?
- A: tự học! (làm việc y như trong trường hợp 1 bcm!)

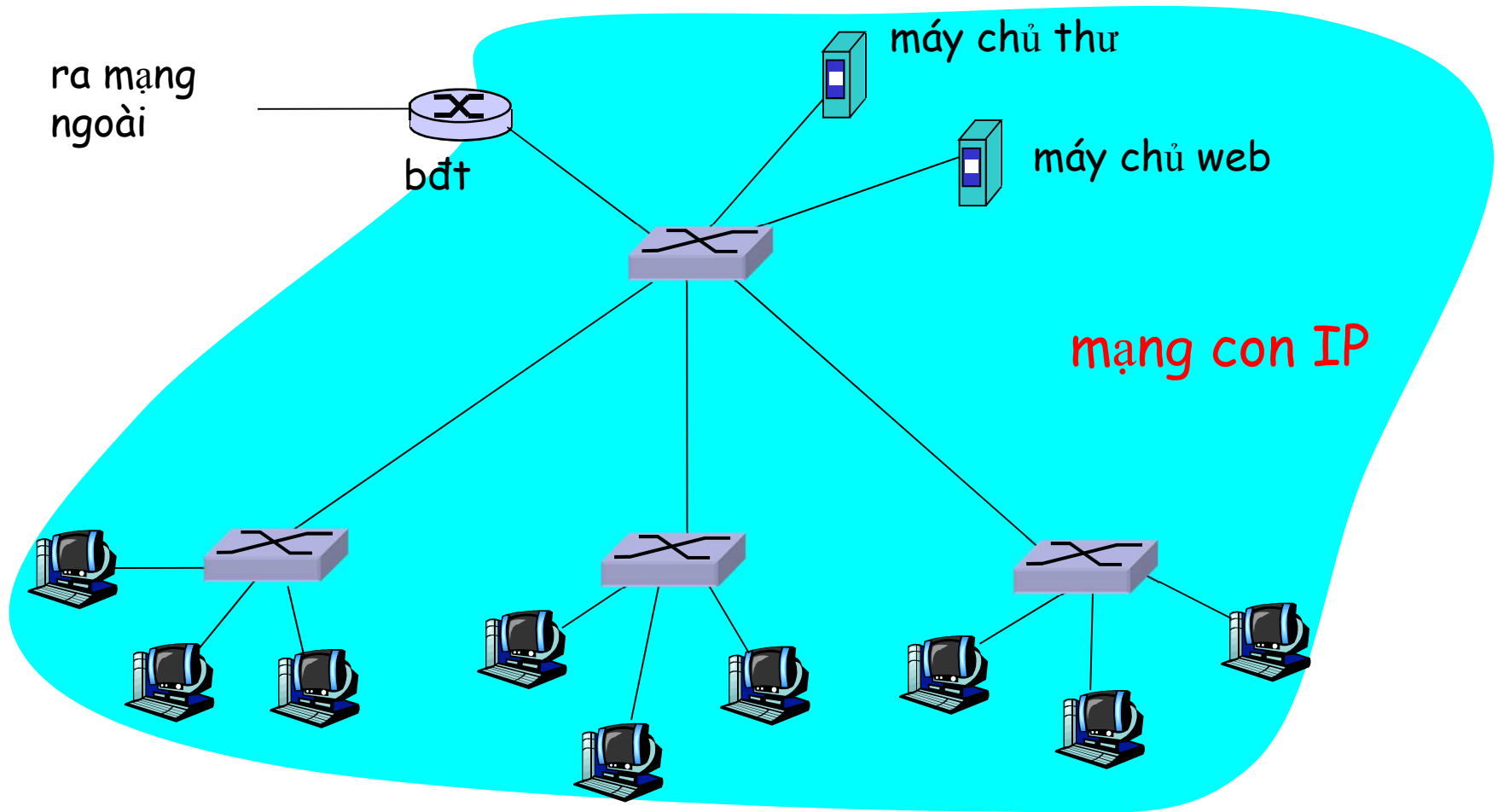
Ví dụ tự-học trong nhiều-bcm

Giả sử C gửi gói cho I, I phản hồi cho C



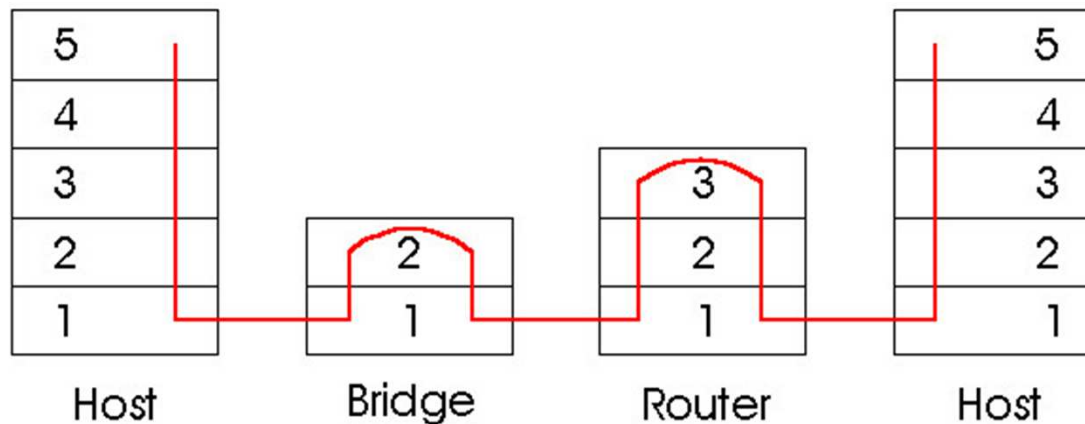
- H: hãy ghi ra bảng định tuyến và sự chuyển tiếp gói trong S_1, S_2, S_3, S_4

Mạng công sở



BCM (switch) so với BĐT (router)

- ❑ cả hai là các thiết bị lưu-và-chuyển tiếp
 - bộ định tuyến: thiết bị tầng mạng (xem xét mào đầu tầng mạng)
 - bộ chuyển mạch là thiết bị tầng liên kết
- ❑ BĐT duy trì bảng định tuyến, thực thi các giao thức định tuyến
- ❑ BCM duy trì bảng chuyển mạch, thực thi các giải thuật lọc và tự học



5: DataLink Layer

Tầng liên kết

- 5.1 Giới thiệu và dịch vụ
- 5.2 Sự phát hiện và Sửa lỗi
- 5.3 Các giao thức đa truy cập
- 5.4 Đánh địa chỉ tầng-Liên kết
- 5.5 Ethernet
- 5.6 Các bộ chuyển mạch tầng-liên kết
- 5.7 PPP

G/thức kiểm soát liên kết dữ liệu điểm-tới-điểm (PPP)

- ❑ 1 n/gửi, 1 n/nhận, 1 liên kết: dễ hơn liên kết quảng bá:
 - không có kiểm soát truy cập môi trường
 - không cần đánh địa chỉ MAC rõ ràng
 - vd: liên kết quay số, đường ISDN
- ❑ các g/thức DLC điểm-tới-điểm phổ biến:
 - PPP (point-to-point protocol)
 - HDLC: High level data link control (tầng liên kết dữ liệu thường được xem là "tầng cao" trong chồng giao thức mạng!)

Các yêu cầu thiết kế PPP [RFC 1557]

- ❑ **chia khung gói tin:** đóng gói gói tin tầng mạng vào trong khung liên kết dữ liệu
 - mang dữ liệu tầng mạng của bất kì giao thức tầng mạng nào (không chỉ IP) ở *cùng thời điểm*
 - khả năng tách các giao thức tầng trên
- ❑ **sự trong suốt bit:** phải mang bất kì mẫu bit nào trong trường dữ liệu
- ❑ **phát hiện lỗi:** (không có sửa lỗi)
- ❑ **sự tồn tại của kết nối:** phát hiện, báo hiệu về sự hỏng liên kết lên tầng mạng
- ❑ **thương lượng địa chỉ tầng mạng:** đầu cuối có thể học/cấu hình địa chỉ mạng của nhau

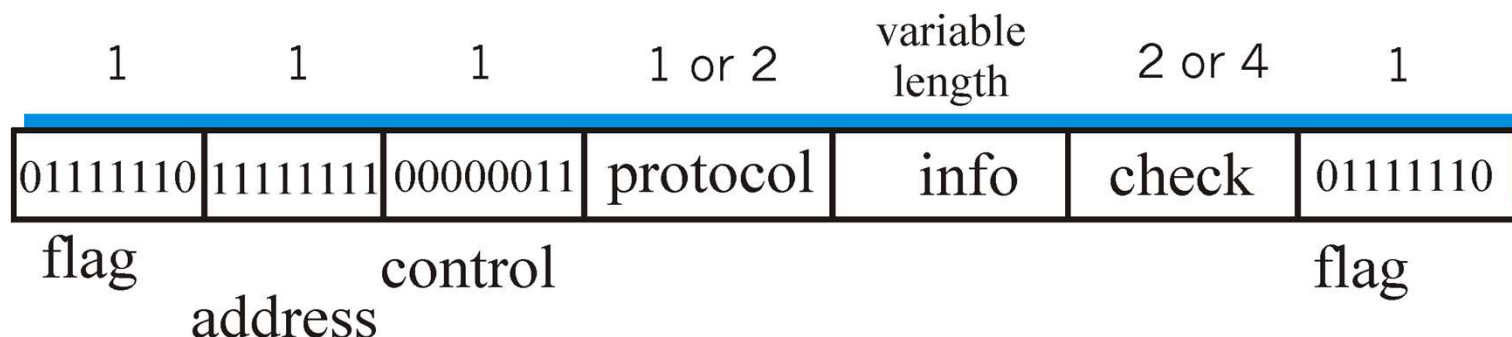
PPP không yêu cầu:

- ❑ sửa/phục hồi lỗi
- ❑ kiểm soát lưu lượng
- ❑ phát đúng thứ tự
- ❑ sự cần thiết phải hỗ trợ liên kết nhiều điểm

Phục hồi lỗi, kiểm soát lưu lượng, xếp trật tự dữ liệu
tất cả đều là công việc của các tầng cao hơn!

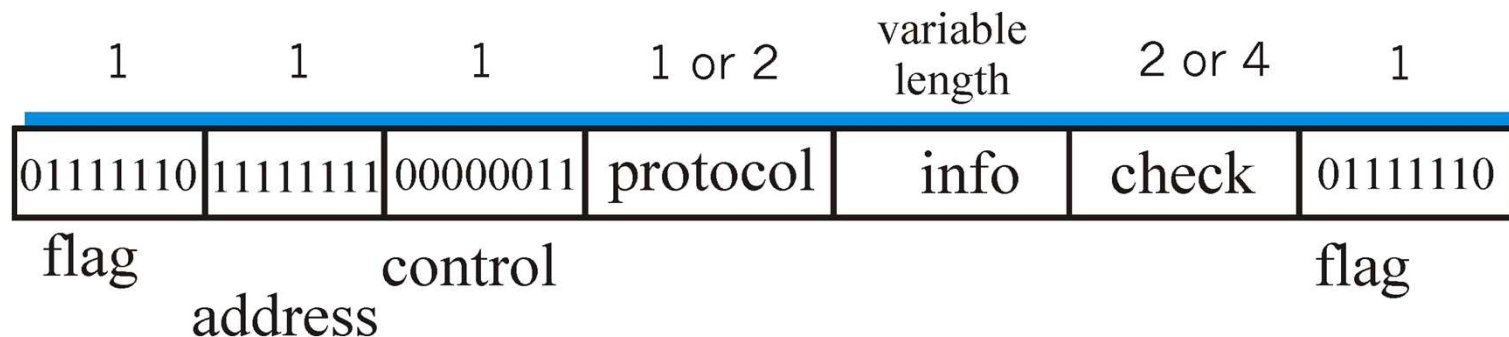
Khung dữ liệu PPP

- ❑ **Cờ:** dấu tách (chia khung)
- ❑ **Địa chỉ:** không có ý nghĩa (chỉ có 1 lựa chọn)
- ❑ **Điều khiển:** không có ý nghĩa; trong tương lai có thể có nhiều trường điều khiển
- ❑ **Giao thức:** giao thức tầng trên mà khung sẽ được đưa tới (vd: PPP-LCP, IP, IPCP, v.v..)



Khung dữ liệu PPP

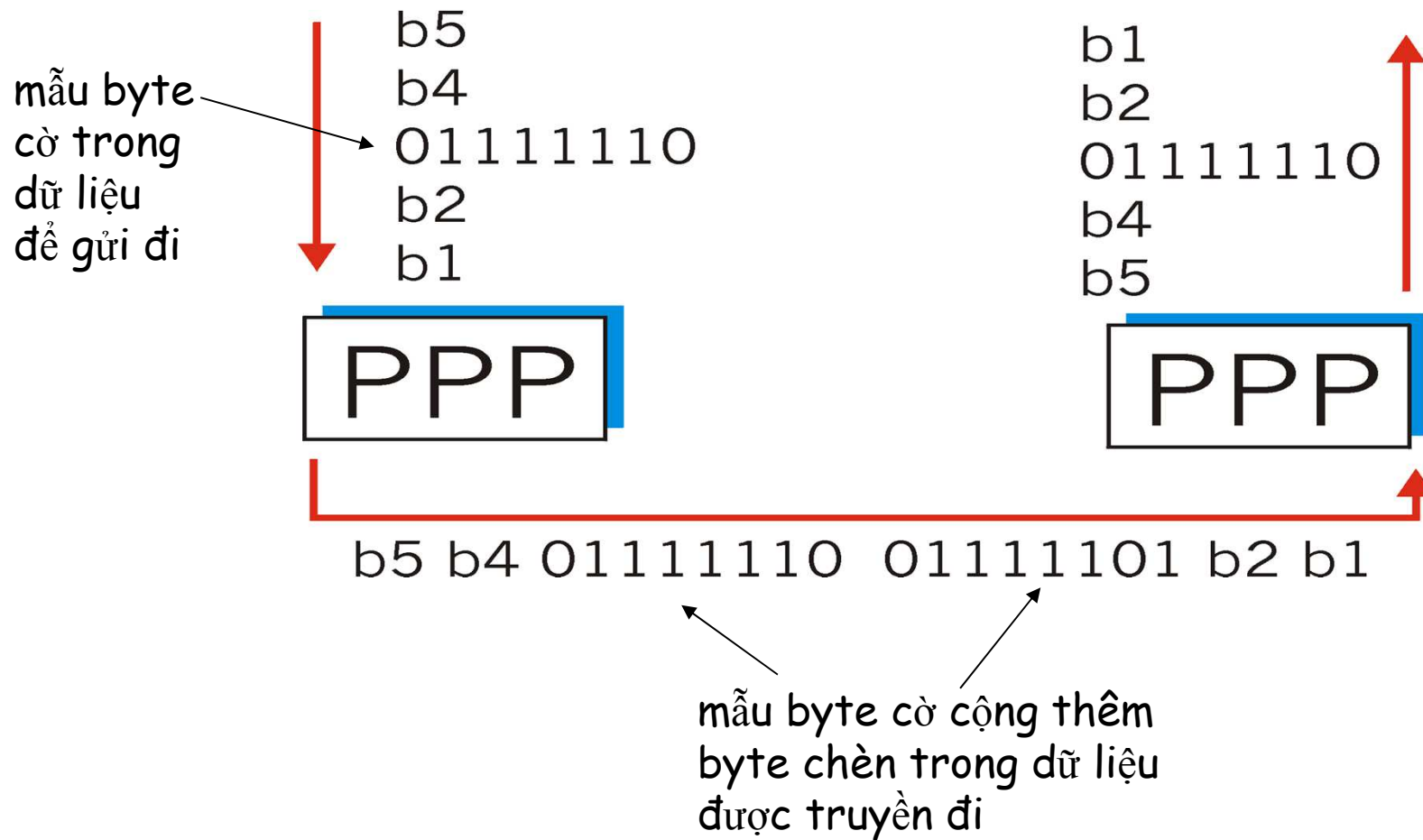
- **thông tin:** dữ liệu tầng cao hơn được mang
- **kiểm tra:** CRC để phát hiện lỗi



Sự chèn Byte

- “sự trong suốt dữ liệu” yêu cầu: trường dữ liệu phải có khả năng chứa mẫu cờ <01111110>
 - H: vậy mẫu <01111110> nhận được là dữ liệu hay cờ?
- N/gửi: thêm (“chèn”) byte < 01111110> thêm vào sau mỗi byte < 01111110> *dữ liệu*
- N/nhận:
 - hai byte 01111110 đi cạnh nhau: bỏ byte đầu và tiếp tục nhận các byte tiếp theo
 - chỉ có 1 byte 01111110: byte cờ

Sự chèn Byte



Giao thức Kiểm soát Dữ Liệu (DLC) PPP

Trước khi trao đổi dữ liệu tầng-mạng, các bên liên kết-dữ liệu phải:

- ❑ **cấu hình liên kết PPP link** (độ dài khung tối đa, xác thực)
- ❑ **học/cấu hình mạng**
thông tin tầng
 - cho IP: mang th/điệp Giao thức Kiểm soát IP (IPCP) (trường protocol: 8021) để cấu hình/học địa chỉ IP

